UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/711,816	10/07/2004	Ching-Hai Hung	IFTP0005USA	5815

27765 7590 03/26/2008

NORTH AMERICA INTELLECTUAL PROPERTY CORPORATION
P.O. BOX 506
MERRIFIELD, VA 22116

EXAMINER	
BIRKHIMER, CHRISTOPHER D	
ART UNIT	PAPER NUMBER
2186	

NOTIFICATION DATE	DELIVERY MODE
03/26/2008	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

winstonhsu.uspto@gmail.com
Patent.admin.uspto.Rcv@naipo.com
mis.ap.uspto@naipo.com.tw

Office Action Summary	Application No.	Applicant(s)	
	10/711,816	HUNG, CHING-HAI	
	Examiner	Art Unit	
	CHRISTOPHER D. BIRKHIMER	2186	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *3* MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *07 October 2004*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1 - 22* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1 - 22* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *07 October 2004* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

This Office Action is in response to application 10711816 submitted 10/07/2004. Claims 1 – 22 are pending in the application.

Claim Rejections - 35 USC § 112

1. The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

2. Claims **1 - 22** are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

Claim 1 recites "wherein after the initialization progress table is created and before the completion of the consistency initialization is completed, the RAID is allowed to access and the consistency initialization is allowed to start. The cited limitation in the claim appears to raise multiple indefinite issues.

a. It is unclear what "the RAID is allowed to access" actually means. This could mean the RAID is allowed to *be* accessed or the RAID itself is allowed to access another device. The Examiner will assume "the RAID is allowed to access" is meant to mean that the RAID is allowed to be accessed. This is supported in the summary of the disclosure and appears to be the main inventive concept by the Applicant. The Applicant states in the prior art that previous RAID consistency initialization processes had to be completed before the RAID was able to be accessed by the computer device it was connected with. It seems the

Applicant's is trying to claim a device that allows a RAID to be accessed while the consistency initialization has no completed yet.

b. It is also unclear what "before completion of the consistency initialization is completed...the consistency initialization is allowed to start." If assumed that if RAID is allowed to be accessed before the consistency initialization is completed that the consistency initialization has already started. The Examiner is unable to determine what “the consistency initialization is allowed to start” actually means in the claim since it is implied above where the RAID is allowed to be accessed before the consistency initialization is completed. The Examiner is going to assume the limitation “before completion of the consistency initialization is completed...the consistency initialization is allowed to start” reads on a initialization consistency starting.

Claim 10 recites that it is dependent on claim 7 and mentions such terms as "the step (a)", "said step (e)", and "steps (a0) through (d). There is no mention in any claims that claim 10 is dependent on of steps (a), (d), or (e). It appears that claim 10 could be dependent from claim 8 or 9, but it is not clear which claim it is meant to be dependent on. There is a lack of antecedent basis for these terms in the claim. The Examiner will assume the steps (a0) through (d) only includes the steps of (a0) and (a) which are included in claim 10 since no other claims are defined and assuming step (e) comes after step (d). The Examiner is also unable to clearly understand what the method of step (a0) performs. It appears it is meant to determine something but from the language of the claim it is unclear what it exactly determines. The Examiner is

interpreting the limitation of step (a0) to mean determining if a region needs to be initialized or not and then selecting that region.

Claim 14 recites the limitation "the regional initialization" in line 3. There is no previous mention of a regional initialization. There is insufficient antecedent basis for this limitation in the claim. The Examiner assumes the regional initialization refers to initialization that is performed on the initialization regions.

Claim 15 recites the limitation "the regional initialization" in line 3 on page 28 and line 1 on page 29. There is no previous mention of a regional initialization. There is insufficient antecedent basis for this limitation in the claim. The Examiner assumes the regional initialization refers to initialization that is performed on the initialization regions.

Claim 16 recites the limitation "the regional initialization" in lines 3 and 5. There is no previous mention of a regional initialization. There is insufficient antecedent basis for this limitation in the claim. The Examiner assumes the regional initialization refers to initialization that is performed on the initialization regions.

Claims 2 - 9, 11 - 13, and 17 - 22 are rejected for being dependent upon a rejected base claim.

Claim Rejections - 35 USC § 103

3. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the

invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

4. The factual inquiries set forth in *Graham* **v.** *John Deere Co.*, 383 U.S. 1, 148 USPQ 459 (1966), that are applied for establishing a background for determining obviousness under 35 U.S.C. 103(a) are summarized as follows:

1. Determining the scope and contents of the prior art.
2. Ascertaining the differences between the prior art and the claims at issue.
3. Resolving the level of ordinary skill in the pertinent art.
4. Considering objective evidence present in the application indicating obviousness or nonobviousness.

5. Claims **1, 3 – 16, and 19 - 21** are rejected under 35 U.S.C. 103(a) as being unpatentable over DeKoning et al (Pat 6,467,023).

With regard to **claim 1**, DeKoning teaches a method for redundant array of independent disks (RAID) **[100, Fig 1; Column 5, Lines 19 - 21]** consistency initialization **[Fig 3 and Fig 4]** comprises:

creating RAID **[100, Fig 1; Column 5, Lines 19 - 21]**, including setting a RAID configuration of the RAID **["configuration information", Column 3, Lines 34 – 41]** and storing progress states of the initialization of the RAID **[300, 302, 304, 306, 307, Fig 3, This describes the function of storing the state of the initialization of the LUN for the RAID system]** wherein before the completion of the consistency initialization **[Fig 3 and Fig 4]** is completed, the RAID **[100, Fig 1; Column 5, Lines 19 – 21]** is allowed to access and the consistency initialization is allowed to start **[Column 3, Lines 41 – 43]**.

However, DeKoning does not specifically disclose the limitation of storing the progress states in a table.

The Examiner is taking Official Notice that it would have been obvious to someone of ordinary skill in the art at the time of the invention to store the progress states in a table structure. There are a finite number of data structures to store data in and a table is one of the well known data structures used to store data. It would have been obvious to use a table since it is well known in the art and by using a table it would allow multiple checkpoints to be maintained allowing the system to return to a particular checkpoint in the LUN initialization if desired **[Column 8, Lines 30 - 37]**.

With regard to **claim 3**, DeKoning teaches the consistency initialization **[Fig 3 and Fig 4]** comprises an induced consistency initialization **[Column 3, Lines 53 – 60]**.

With regard to **claim 4**, DeKoning teaches the induced consistency initialization **[Fig 4; Column 3, Lines 53 – 60]** comprises:

(a). detecting whether the consistency initialization is completely performed on the whole RAID **[100, Fig 1; Column 5, Lines 19 – 21]** when the RAID **[100, Fig 1; Column 5, Lines 19 – 21]** receives an I/O **[402, Fig 4, Checking if one of the LUN is initialized is the same as checking if the whole RAID is initialized since all the LUN need to be initialized for the whole RAID to be initialized]**;

(b). if step (a) is negative, detecting whether a regional initialization **["PORTION", 300, 302, 304, 306, 307, 308, and 310, Fig 3]** is completely performed on an initialization region **["PORTION", 300 and 310, Fig 3]** associated with the I/O **[404, Fig 4, If it is above the boundary then the**

regional initialization is completed and if it below the boundary then the regional initialization is not completely performed];

(c). if step (b) is negative, detecting whether the regional initialization **["PORTION", 300, 302, 304, 306, 307, 308, and 310, Fig 3]** is being performed on said initialization region **["PORTION", 300 and 310, Fig 3]** associated with the I/O **[405, Fig 4; Column 9, Lines 9 – 21, If the request is above and below the boundary that indicates that the regional initialization has started but is not complete for the region the I/O is associated with]**;

(d). if step (c) is affirmative, waiting for the completion of the regional initialization **["PORTION", 300, 302, 304, 306, 307, 308, and 310, Fig 3; 407, Fig 4; Column 9, Lines 21 – 23]**, and if step (c) is negative, performing the regional initialization **["PORTION", 300, 302, 304, 306, 307, 308, and 310, Fig 3]** on said initialization region **["PORTION", 300 and 310, Fig 3]** and updating initialization a state change of the initialization region **["PORTION", 300 and 310, Fig 3]** to said initialization progress table **[300, 302, 304,, 306, 307, 308, and 310, Fig 3; 405 and 408, Fig 4; The flow chart of Fig 3 shows that once initialization starts on a LUN all the regions will eventually get initialized. The flow chart of Fig 4 shows that initialization of a region can be detected as not happening yet if the flow chart reaches step 408. Even if step 408 in**

Fig 4 is reached the process of Fig 3 is still running so the initialization of the region below the boundary will occur];

(e). writing the updated initialization progress table into the non-volatile memory device **[118, Fig 1; Column 5, Lines 64 – 67; Column 6, Lines 1 – 5]** if a predetermined condition is met **[304, 306, and 307, Fig 3]**.

With regard to **claim 5**, DeKoning teaches wherein the I/O access the RAID **[100, Fig 1; Column 5, Lines 19 – 21]** after step e **[Column 7, Lines 12 – 23, This shows the I/O accesses the RAID before and after a particular update to the progress table since the initialization and I/O accesses are performed in parallel]**.

With regard to **claim 6**, DeKoning teaches wherein the I/O access the RAID **[100, Fig 1; Column 5, Lines 19 – 21]** before step e **[Column 7, Lines 12 – 23, This shows the I/O accesses the RAID before and after a particular update to the progress table since the initialization and I/O accesses are performed in parallel]**.

With regard to **claim 7**, DeKoning teaches the consistency initialization **[Fig 3 and Fig 4]** further comprises a consecutive consistency initialization **[Fig 3, The figure shows that initialization starts at the beginning of the LUN and ends once it has initialized the whole LUN]**.

With regard to **claim 8**, DeKoning teaches the consecutive consistency initialization **[Fig 3, The figure shows that initialization starts at the beginning of the LUN and ends once it has initialized the whole LUN]** comprises following steps:

(a). selecting an initialization region **["PORTION", 300 and 310, Fig 3]** which has not been completed with initialization yet **[310, Fig 3]**;

(b). if a regional initialization **["PORTION", 300, 302, 304, 306, 307, 308, and 310, Fig 3]** is not being performed on the selected initialization region **["PORTION", 300 and 310, Fig 3]**, performing the regional initialization **["PORTION", 300, 302, 304, 306, 307, 308, and 310, Fig 3]** on the initialization region **[308, 310, and 302, Fig 3]**;

(c). if step (b) is performed, updating the initialization state changes of the initialization region to an initialization progress table **[307, Fig 3; As stated above in the rejection of claim 1, it would be obvious to store the checkpoint save into a table which would provide the ability to recover the system to a certain point of initialization if there was a failure during the initialization]**;

(d). if a second predetermine condition is met **[304, 306, and 307, Fig 3]**, writing the updated initialization progress table into the memory device **[118, Fig 1; Column 5, Lines 64 - 67; Column 6, Lines 1 - 5]**;

(e). repeating steps (a) through (d) until initialization regions **["PORTION", 300 and 310, Fig 3]** are completed with initialization **[308 and 312, Fig 3]**.

With regard to **claim 9**, DeKoning teaches:

(f). writing a state showing that all initialization regions **["PORTION", 300 and 310, Fig 3]** are completed with initialization **[312, Fig3]** into a non-volatile memory device after step (e) **[118, Fig 1; Column 5, Lines 64 – 66; Column 6, Lines 1 – 5; As stated above in the rejection of**

claim 2, it is implied that the memory that initialization information is stored in is non-volatile memory or else the initialization would need to be run every time the system is powered down and then back up].

With regard to **claim 10**, DeKoning teaches (a0) performing a regional initialization priority adjustment mechanism to determine whether selecting an initialization region **["PORTION", 300 and 310, Fig 3]** which has not yet been completed with initialization or not before **[308, Fig 3]** the step (a) selecting the initialization region **["PORTION", 300 and 310, Fig 3]** which has not been completed with initialization yet **[310, Fig 3]** and said step (e) comprising repeating steps (a0) **[308, Fig 3]** through (d) until all initialization regions **["PORTION", 300 and 310, Fig 3]** have been completed with initialization **[308 and 312, Fig 3, This shows initialization is processed until the whole LUN is initialized]**.

With regard to **claim 11**, DeKoning teaches the consistency initialization **[Fig 3 and Fig 4]** further comprises a consecutive consistency initialization **[Fig 3, The figure shows that initialization starts at the beginning of the LUN and ends once it has initialized the whole LUN]** and after the initialization progress table is created, the consecutive consistency initialization **[Fig 3, The figure shows that initialization starts at the beginning of the LUN and ends once it has initialized the whole LUN]** is allowed to start anytime **["CHECKPOINT RESTART" and 316, Fig 3; Column 8, Lines 30 – 38, This shows that once the initialization progress table is created the consecutive consistency initialization is allowed to start]**

With regard to **claim 12**, DeKoning teaches the RAID **[100, Fig 1; Column 5, Lines 19 - 21]** is allowed I/O accessing **[Column 3, Lines 50 – 53]** before the consecutive consistency initialization **[Fig 3]**.

With regard to **claim 13**, DeKoning teaches the consistency initialization **[Fig 3 and Fig 4]** comprises dividing a data space of member disks **[108, Fig 1]** into a plurality of initialization regions **["PORTION", 300 and 310, Fig 3]** and performing the regional initialization **["PORTION", 300, 302, 304, 306, 307, 308, and 310, Fig 3]** on the initialization regions **["PORTION", 300, 302, 304, 306, 307, 308, and 310, Fig 3]**.

With regard to **claim 14**, DeKoning teaches the consistency initialization **[Fig 3 and Fig 4]** comprises dividing a data space of member disks **[108, Fig 1]** into a plurality of initialization regions **["PORTION", 300 and 310, Fig 3]** and performing the regional initialization on the initialization regions **["PORTION", 300, 302, 304, 306, 307, 308, and 310, Fig 3]**.

With regard to **claim 15**, DeKoning teaches the consistency initialization **[Fig 3 and Fig 4]** comprises dividing a data space of member disks **[108, Fig 1]** into a plurality of initialization regions **["PORTION", 300 and 310, Fig 3]** and performing the regional initialization on the initialization regions **["PORTION", 300, 302, 304, 306, 307, 308, and 310, Fig 3]**, and the initialization progress updated due to an I/O **["INITIALIZE LUN", Fig 3]** accessing a data space of the RAID **[100, Fig 1; Column 5, Lines 19 – 21; 307, Fig 3, The initialization of the RAID is an I/O access since a RAID system is an I/O device]** and inducing the regional initialization **[302, Fig 3, The initialization is induced since the current portion of the LUN is an un-initialized portion that is**

accessed by an I/O access since the RAID device is an I/O device] is written in to the non-volatile memory device **[118, Fig 1; Column 5, Lines 64 – 66; Column 6, Lines 1 – 5]** after said accessing has been completed **[307, Fig 3]**, and then the I/O **["INITIALIZE LUN", Fig 3]** result is returned **[312, Fig 3]**.

However, DeKoning does not specifically disclose the limitation of storing the progress in a table.

The Examiner is taking Official Notice that it would have been obvious to someone of ordinary skill in the art at the time of the invention to store the progress in a table structure. There are a finite number of data structures to store data in and a table is one of the well known data structures used to store data. It would have been obvious to use a table since it is well known in the art and by using a table it would allow multiple checkpoints to be maintained allowing the system to return to a particular checkpoint in the LUN initialization if desired **[Column 8, Lines 30 - 37]**.

With regard to **claim 16**, DeKoning teaches the consistency initialization **[Fig 3 and Fig 4]** comprises dividing a data space of member disks **[108, Fig 1]** into a plurality of initialization regions **["PORTION", 300 and 310, Fig 3]** and performing the regional initialization on the initialization regions **["PORTION", 300, 302, 304, 306, 307, 308, and 310, Fig 3]** and the initialization progress updated due to an I/O **["INITIALIZE LUN", Fig 3]** and inducing the regional initialization **[302, Fig 3, The initialization is induced since the current portion of the LUN is an un-initialized portion that is accessed by the I/O access since the RAID device is an I/O device]** is written in to the non-volatile memory device **[118, Fig 1; Column 5, Lines 64 – 66; Column 6,**

Lines 1 – 5] first, and then said I/O **["INITIALIZE LUN", Fig 3]** accessing a data space of the RAID **[100, Fig 1; Column 5, Lines 19 – 21; 308 and 310, Fig 3, The I/O accesses a all the data spaces until the entire LUN is initialized]**.

With regard to **claim 19**, DeKoning teaches if a member disk failed and a new member disk **[Column 9, Lines 60 – 62]** is used to perform a rebuilding of the RAID **[100, Fig 1; Column 5, Lines 19 - 21]** before the completion of the consistency initialization **[Column 9, Lines 55 – 58]**, the rebuilding only has to perform on the regions which have been completed with the consistency initialization and the rebuilding on the regions which have not been completed with the consistency initialization can be performed by the consistency initialization **[308 and 312, Fig 3; 502 and 504, Fig 5; Column 10, Lines 1 – 11, This shows the rebuilding is done on the redundancy information affected by the replace which would only include initialized regions and then the LUN is marked as fully initialized once initialization is complete]**.

With regard to **claim 20**, DeKoning teaches when an I/O operation **["PROCESS I/O REQUEST", Fig 4]** accesses the RAID **[100, Fig 1; Column 5, Lines 19 - 21]** is a read operation **[The I/O request indicates both read and write requests]** and a region of the RAID **[100, Fig 1; Column 5, Lines 19 - 21]** to be accessed by the I/O has not been initialized yet **[405 and 407, Fig 4]**, no consistency initialization is performed on the region **[407, Fig 4]**, and a value of zero or a predetermined value will be returned directly **[407, Fig 3, The I/O request is a predetermined value since it is the same I/O request at the beginning of the process and not a newly determined I/O**

request. The predetermined I/O request is returned directly to a queue to be processed later].

With regard to **claim 21**, DeKoning teaches a RAID **[100, Fig 1; Column 5, Lines 19 - 21]** performs an I/O operation **["PROCESS I/O REQUEST", Fig 4]** and causes an induced consistency initialization **[408, Fig 4; Column 9, Lines 11 – 18, The claim does not require that the updated initialization will not be written into the memory disks again due to a completion of the I/O operation since it is dependent on a particular outcome of a conditional statement. If the particular outcome is not met then the limitation of the updated initialization will not be written into the memory disks again due to a completion of the I/O operation is not required and DeKoning reads on the other outcome of the conditional statement not listed in the claim]**.

6. Claims **2 and 17 - 18** are rejected under 35 U.S.C. 103(a) as being unpatentable over DeKoning et al (Pat 6,467,023) as applied to claim **1** above, and further in view of Humlicek et al (Pat 5,822,782).

With regard to **claim 2**, DeKoning teaches the RAID configuration is stored in a random access memory **[118, Fig 1; Column 5, Lines 64 – 66; Column 6, Lines 1 – 5]**.

However, DeKoning does not specifically disclose the limitation that the memory is a non-volatile memory device.

Humlicek discloses a RAID system that stores configuration information on a non-volatile memory device **[Column 6, Lines 52 - 62]**.

It would have been obvious to someone of ordinary skill in the art at the time of the invention to use the teachings of Humlicek in DeKoning, because it permits improved flexibility in the use of the RAID system by allowing by allowing drives to be physically removed without requiring a rebuild of the stored data **[Column 3, Lines 19 – 26]**.

With regard to **claim 17**, DeKoning teaches the RAID configuration is stored in a random access memory **[118, Fig 1; Column 5, Lines 64 – 66; Column 6, Lines 1 – 5]**.

However, DeKoning does not specifically disclose the limitation that the non-volatile memory is a member disk.

Humlicek discloses a RAID system that stores configuration information on the disk drives **[Column 6, Lines 52 - 62]**.

With regard to **claim 18**, DeKoning teaches there are a plurality of versions of the initialization progress **[307, Fig 3, Each save would result in different version of data]** stored in the memory device **[118, Fig 1; Column 5, Lines 64 – 66; Column 6, Lines 1 – 5]**

However, DeKoning does not specifically disclose the limitation of storing the progress in a table.

The Examiner is taking Official Notice that it would have been obvious to someone of ordinary skill in the art at the time of the invention to store the progress

states in a table structure. There are a finite number of data structures to store data in and a table is one of the well known data structures used to store data. It would have been obvious to use a table since it is well known in the art and by using a table it would allow multiple checkpoints to be maintained allowing the system to return to a particular checkpoint in the LUN initialization if desired **[Column 8, Lines 30 - 37]**.

7. Claim **22** is rejected under 35 U.S.C. 103(a) as being unpatentable over DeKoning et al (Pat 6,467,023) in view of Humlicek et al (Pat 5,822,782) as applied to claim **2** above, and further in view of Rezaul Islam et al (Pat 6,282,670).

With regard to **claim 22**, DeKoning teaches the RAID configuration is stored in a random access memory **[118, Fig 1; Column 5, Lines 64 – 66; Column 6, Lines 1 – 5]**.

Humlicek discloses a RAID system that stores configuration information on a non-volatile memory device **[Column 6, Lines 52 - 62]**.

However, DeKoning in view of Humlicek does not specifically disclose the limitation that the memory device is a battery backed-up SRAM, a flash RAM, or a disk drive except a memory disk.

Rezaul Islam discloses storing configuration data in a flash memory **[128, Fig 1; Column 8, Lines 39 – 45]**.

It would have been obvious to someone of ordinary skill in the art at the time of the invention to use the teachings of Rezaul Islam in DeKoning in view of Humlicek, because flash memory is inexpensive and easily reprogrammable.

Conclusion

8. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

Duncan et al (Pub. No.: US 2003/0140299 A1) discloses when there is no initialization pattern present an error message is returned **[Page 4, Paragraph 0036, Lines 16 – 24]**.

Matsunami et al (Pub. No.: US 2002/0099914 A1) discloses storing information about a RAID configuration in a table format **[Fig 9]**.

Islam et al (Pat 5,950,230) discloses storing RAID configuration data in non-volatile memory and flash memory **[Column 4, Lines 5 – 7]**.

Gentry et al (Pat 5,568,629) discloses logical configuration of a disk array stored in several tables **[Column 5, Lines 8 – 9]**.

Galbraith et al (Pat 5,537,567) discloses storing configuration information in RAM and flash memory **[Column 5, Lines 39 – 41]**.

Hewlett Packard (White Paper: Flash Dimm Technology) discloses the advantages of flash RAM memory.

Direction Of Future Correspondence

Any inquiry concerning this communication or earlier communications from the examiner should be directed to CHRISTOPHER D. BIRKHIMER whose telephone number is (571)270-1178. The examiner can normally be reached on M-H 7:00 - 5:00.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Matt Kim can be reached on 571-272-4182. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Christopher D Birkhimer
Examiner
Art Unit 2186

/Christopher D Birkhimer/
Examiner, Art Unit 2186

/Tuan V. Thai/
Primary Examiner, Art Unit 2186

Deleted: ¶

Deleted: ¶
¶